

วิชา การวิเคราะห์และออกแบบเชิงวัตถุ

(Object Oriented Analysis and Design)



ผู้ช่วยศาสตราจารย์จุฑาวุฒิ จันทรมาลี

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสวนดุสิต

หน่วยที่ 7

การออกแบบ Operation และแผนภาพ Interaction Diagram

สาระการเรียนรู้

1. การออกแบการทำงาน (Operation)
2. สัญลักษณ์ทั่วไปของ UML
3. วัตถุประสงค์ของ Use Case Diagram
4. ประโยชน์ของ Use Case Diagram
5. ขั้นตอนการสร้าง Use Case Diagram

สาระการเรียนรู้

6. การปฏิสัมพันธ์ (Interaction)
7. ความหมายของ Interaction Diagrams
8. วัตถุประสงค์ของ Interaction Diagrams
9. วิธีการสร้างแผนภาพ Interaction Diagrams แบบ Sequence Diagram
10. ขั้นตอนการสร้าง Sequence Diagram

จุดประสงค์การเรียนรู้

1. บอกความหมายของการออกแบบการทำงานได้
2. อธิบายวัตถุประสงค์และประโยชน์ของ Use Case Diagram ได้
3. อธิบายความหมายและวัตถุประสงค์ของ Interaction Diagrams ได้
4. อธิบายวิธีการสร้างแผนภาพ Interaction Diagrams แบบ Sequence Diagram ได้
5. ประยุกต์การออกแบบ Operation และแผนภาพ Interaction Diagram ได้

สมรรถนะการเรียนรู้

1. แสดงความรู้เกี่ยวกับการออกแบบการทำงาน
2. แสดงความรู้เกี่ยวกับ Use Case Diagram และ Interaction Diagrams
3. ปฏิบัติการออกแบบ Operation และแผนภาพ Interaction Diagram

การออกแบบการทำงาน (Operation)

การออกแบบการทำงาน เป็นกระบวนการกำหนดงานเฉพาะอย่างที่จะต้องทำ วิธีการที่ใช้ในการทำงาน และวิธีการที่เกี่ยวข้องกับงานอื่น ๆ ในองค์กร หรือเป็นกระบวนการของ การกำหนดงานของพนักงานตามลักษณะ โครงสร้างให้สอดคล้องกับลักษณะของบุคคลเพื่อให้ บรรลุผลสำเร็จตามวัตถุประสงค์ขององค์กร

กระบวนการออกแบบงาน

1. ลักษณะเฉพาะอย่างของแต่ละงาน (the specification of individual tasks) หมายถึง งานที่แตกต่างที่พนักงานแต่ละคนทำ
2. ลักษณะเฉพาะอย่างของวิธีการปฏิบัติงาน (the specification of the method of performing each task) หมายถึง งานแต่ละงานมีวิธีการทำอย่างไร
3. การรวมแต่ละงานให้เป็นงานเฉพาะของแต่ละคน (the combination of individual tasks into specific jobs to be assigned to individuals) หมายถึง วิธีการทำงานในหน้าที่ที่แตกต่าง ได้รับการรวบรวมขึ้นเพื่อสร้างรูปแบบงาน

วัตถุประสงค์ของ Use Case Diagram

1. อธิบายเรื่องราวของ Problem Domain ทั้งหมด (Domain คือ กรอบหรือขอบเขตที่สนใจ)
2. บอกส่วนประกอบในระบบ (ระบบประกอบด้วยระบบย่อยอะไรบ้าง)
3. บอกความสัมพันธ์ของส่วนต่าง ๆ ในระบบ

ประโยชน์ของ Use Case Diagram

1. ช่วยให้ผู้พัฒนาระบบสามารถแยกแยะกิจกรรมที่อาจจะเกิดขึ้นในระบบ
2. เป็น Diagram พื้นฐาน ที่สามารถอธิบายสิ่งต่าง ๆ ได้โดยใช้รูปภาพที่ไม่ซับซ้อน
3. Use Case Diagram จะมีประสิทธิภาพ หากผู้เขียนมีความเข้าใจใน Problem Domain อย่างแท้จริง

ขั้นตอนการสร้าง Use Case Diagram

1. ค้นหา Actor
2. ค้นหา Use Case ที่มีปฏิสัมพันธ์กับ Actor นั้นโดยตรง
3. ค้นหาและสร้างความสัมพันธ์ระหว่าง Use Case หรือ Actor (ถ้ามี) แล้วเพิ่มเติม Use Case ใหม่ ซึ่ง อาจเป็น Included Use Case, Extending Use Case ที่เพิ่มเติมจาก Base Use Case ที่มีอยู่แล้ว หรือจะเพิ่ม Base Use Case ใหม่ก็ได้ (ถ้ามี)

ขั้นตอนการสร้าง Use Case Diagram

4. ต้องไม่มี Actor ใดเลยที่ไม่มีปฏิสัมพันธ์กับ Use Case
5. ต้องไม่มี Use Case ใดเลยที่ไม่มีปฏิสัมพันธ์กับ Actor
6. Use Case ทุกตัวต้องมีปฏิสัมพันธ์อย่างใดอย่างหนึ่งกับ Actor หรือ Use Case ตัวอื่น ๆ เสมอ
7. เขียนคำอธิบายแต่ละ Use Case จนครบถ้วน

การปฏิสัมพันธ์ (Interaction)

การปฏิสัมพันธ์ (Interaction) คือ การสื่อสารระหว่างผู้ใช้กับระบบ โดยที่ระบบมีส่วนต่อประสานเป็นทั้งส่วนที่ผู้ใช้สนใจและเป็นเหมือนคน สันทนา/ตัวกลางระหว่างผู้ใช้และระบบ เริ่มจากผู้ใช้ป้อนคำสั่ง/ออกคำสั่งแก่ ส่วนต่อประสานจากนั้นเป็นหน้าที่ของส่วนต่อประสานที่จะดำเนินการตามคำสั่ง ดังนั้นการสื่อสารระหว่างผู้ใช้และระบบมีความหมายคือเป็นภาษาทางอ้อม (Indirect language) แทนที่จะเป็นการกระทำโดยตรง (Direct Action)

ความหมายของ Interaction Diagrams

Interaction Diagrams คือ แผนภาพที่อธิบายการโต้ตอบระหว่างองค์ประกอบต่าง ๆ ในแบบจำลอง การโต้ตอบนี้เป็นส่วนหนึ่งของพฤติกรรมแบบไดนามิกซ์ของระบบ วัตถุประสงค์พื้นฐานของไดอะแกรมทั้งสองมีความคล้ายคลึงกัน แผนภาพลำดับจะเน้นที่ลำดับเวลาของข้อความและแผนภาพการทำงานร่วมกันจะเน้นที่การจัดโครงสร้างของวัตถุที่ส่งและรับข้อความ

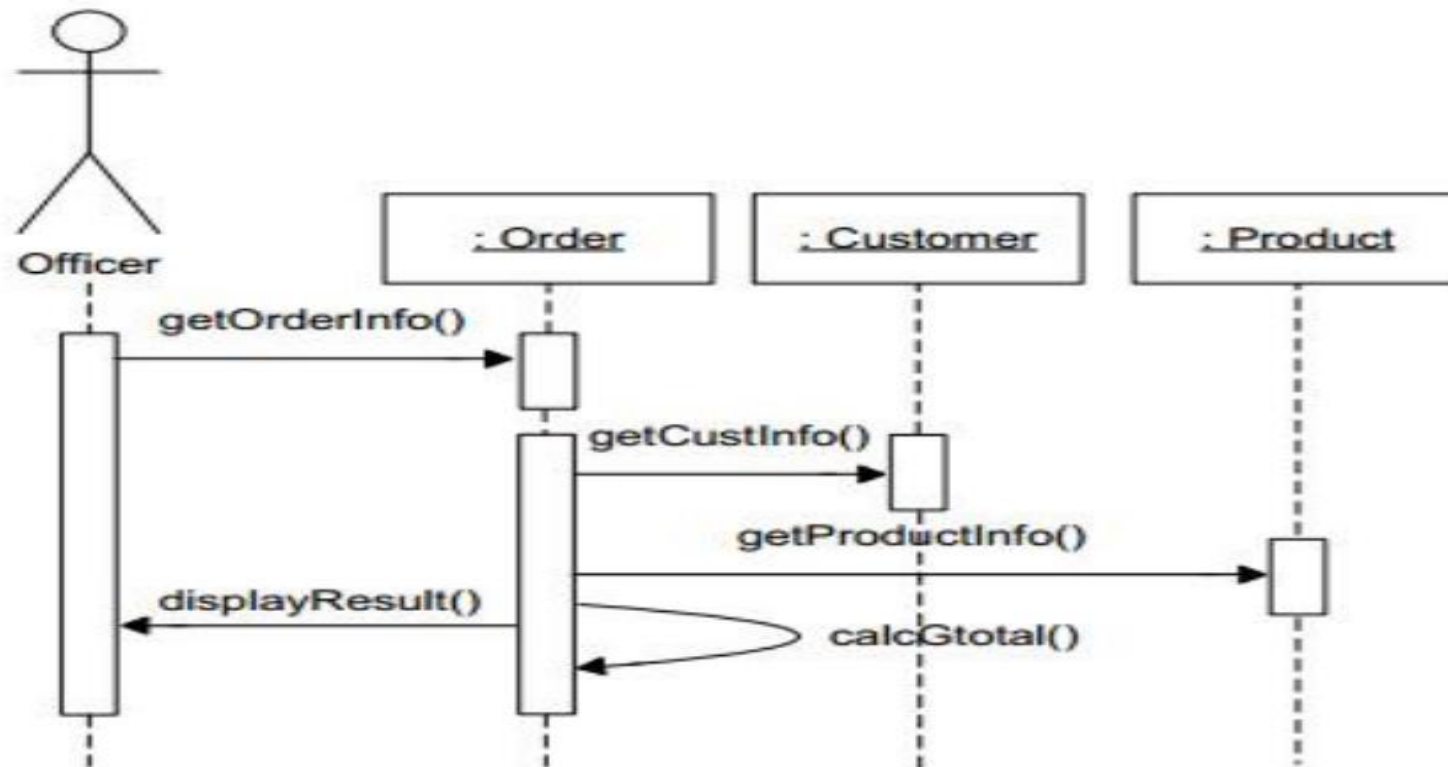
ความหมายของ Interaction Diagrams

1. Call หมายถึง Message ที่ Sender ใช้เรียก Method ของ Receiver
2. Return หมายถึง Message ที่ใช้เพื่อส่งข้อมูลที่ถูกร้องขอโดย Sender จาก Receiver กลับไปยัง Sender
3. Send หมายถึง สัญญาณบางอย่างที่ Object ตัวหนึ่งส่งไปเพื่อบอกหรือกระตุ้น Object อีกตัวหนึ่ง โดยไม่ใช่ การเรียกใช้ Method ของ Object ที่ถูกกระตุ้น
4. Create หมายถึง Message ที่ Object ตัวหนึ่งส่งไปโดยมีจุดประสงค์เพื่อให้เกิดการสร้าง Object ของ Class ขึ้น
5. Destroy หมายถึง Message ที่ Object ตัวหนึ่ง ส่งไปยัง Object อีกตัวหนึ่ง เพื่อให้ Object ที่ได้รับ Message ทำลายตัวเอง

วัตถุประสงค์ของ Interaction Diagrams

วัตถุประสงค์ของ Interaction Diagrams คือการเห็นภาพพฤติกรรม การโต้ตอบของระบบ การมองเห็นการโต้ตอบเป็นเรื่องยาก ดังนั้นการแก้ปัญหาคือการใช้รูปแบบต่าง ๆ เพื่อจับภาพแง่มุมต่าง ๆ ของการมีปฏิสัมพันธ์ แผนภาพ Sequence diagram และ Collaboration diagram จะใช้ในการจับภาพธรรมชาติแบบใดนามิกแต่จากมุมมองที่แตกต่างกัน

วิธีการสร้างแผนภาพ Interaction Diagrams แบบ Sequence Diagram



จบการนำเสนอ

